

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Blaž Oblak

Mobilna aplikacija za ljudi z bralnimi težavami

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

Ljubljana, 2014

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Blaž Oblak

Mobilna aplikacija za ljudi z bralnimi težavami

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: viš. pred. dr. Alenka Kavčič
SOMENTOR: doc. Domen Fras

Ljubljana, 2014

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Mobilna aplikacija za ljudi z bralnimi težavami

Tematika naloge:

V okviru diplomske naloge izdelajte mobilno aplikacijo, ki poskuša pomagati ljudem z bralnimi težavami preko individualnih nastavitev prikaza besedila. Osredotočite se predvsem na disleksijo in možne nastavitve v aplikaciji prilagodite tej bralni motnji. Aplikacija naj nudi podporo pri branju dokumentov pdf ter poljubnih spletnih strani. Raziščite primernost razvoja domorodne aplikacije za dve najpogostejši mobilni platformi, Android in iOS, ter ga primerjajte z razvojem hibridne aplikacije z orodjem PhoneGap.

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Blaž Oblak, sem avtor diplomskega dela z naslovom:

Mobilna aplikacija za ljudi z bralnimi težavami

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom viš. pred. dr. Alenke Kavčič in somentorstvom doc. Domna Frasa,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 30. oktobra 2014

Podpis avtorja:

Zahvala

Zahvaljujem se svoji mentorici viš. pred. dr. Alenki Kavčič za mentorstvo, somentorju doc. Domnu Frasu ter svoji diplomski sodelavki Urški Stariha. Posebna zahvala pa Ireni Hozjan za lektoriranje.

Kazalo

Povzetek

Abstract

1 Uvod	1
2 Disleksija	3
2.1 Priporočila za disleksike	4
2.1.1 Tekst	4
2.1.2 Govorni vmesnik.....	5
2.1.3 Spletno načrtovanje	6
2.2 Pisava.....	6
3 Razvoj aplikacije	9
3.1 Android.....	10
3.2 PhoneGap.....	11
3.3 iOS	14
3.3.1 Razvojno okolje Xcode	14
3.3.2 Objective-C.....	15
3.3.3 Uporabniški vmesnik	16
4 Aplikacija za pomoč pri branju.....	17
4.1 Uvoz dokumenta pdf.....	18
4.2 Razčlenjevalnik pdf.....	19
4.3 Govorni vmesnik.....	19
4.4 Zavihek Library	23
4.4.1 Nastavitve za branje	26
4.5 Zavihek Web.....	28
4.5.1 Spletne nastavitve	31
4.6 Zavihek Settings.....	32
4.7 Zavihek About.....	33
5 Primerjava razvitih aplikacij.....	35
6 Sklepne ugotovitve	39

Seznam uporabljenih kratic

kratica	angleško	slovensko
PDF	Portable Document Format	format datoteke za od platforme neodvisne predstavitev dokumentov
TTS	Text to Speech	pretvornik teksta v govor (govorni vmesnik)
HTTP	HyperText Transfer Protocol	protokol za izmenjavo hiperteksta ter grafičnih, zvočnih in drugih večpredstavnostnih vsebin na spletu
CSS	Cascading Style Sheets	slogovna predloga na spletni strani, v kateri je zapisana oblika spletne strani
HTML	Hyper Text Markup Language	označevalni jezik za oblikovanje večpredstavnostnih dokumentov, ki omogoča povezave znotraj dokumenta ali med dokumenti
API	Application Programming Interface	vmesnik, ki zagotavlja, da ima računalniški program na razpolago funkcije operacijskega sistema ali drugega računalniškega programa
IDE	Integrated Development Environment	integrirano razvojno okolje (programski paket, namenjen programiranju, ki navadno obsega urejevalnik besedil, prevajalnik, povezovalnik in iskalnik napak)
GUI	Graphical User Interface	grafični uporabniški vmesnik
URL	Uniform Resource Locator	enolični naslov spletnega vira

Povzetek

Namen diplomske naloge je izdelati mobilno aplikacijo, ki bo v pomoč osebam z bralnimi težavami. Osredotočili smo se na osebe z disleksijo. Najprej smo preučili disleksijo ter načine kako olajšati simptome in nuditi pomoč bralcu. S pomočjo sodelavcev z Naravoslovnotehniške fakultete ter z raziskavo že obstoječih aplikacij smo si zadali cilje glede funkcionalnosti naše aplikacije. Naš prvi cilj je, da bo uporabnik aplikacije lahko bral e-knjige v formatu pdf z nastavitvami, ki mu bodo olajšale branje. Drugi cilj pa je, da bo uporabnik lahko brskal po spletu, kjer mu bodo nastavitve v aplikaciji priredile spletno stran na način, ki bo omogočal lažji ogled spletne strani za osebe z disleksijo. Odločili smo se, da bomo izdelali aplikacijo za operacijski sistem Android, vendar smo imeli težave pri razčlenjevanju dokumenta pdf. Poizkusili smo tudi z izgradnjo na platformi PhoneGap, vendar smo tudi tukaj imeli težave. Na koncu smo vse zadane cilje uspeli implementirati za operacijski sistem iOS. Aplikacija bo dosegljiva brezplačno, na spletni trgovini App Store.

Ključne besede: disleksija, Android, PhoneGap, iOS, mobilna aplikacija

Abstract

The purpose of the thesis is to develop a mobile application that will help people with reading disorder. We focused on people with dyslexia. First, we examined dyslexia and ways to ease symptoms, that will help the reader. With the help of colleagues from the Faculty of Natural Sciences and with study of existing applications, we have set goals regarding the functionality of our applications. Our first goal was to make the application, that you can read e-books in pdf format, with settings that will make it easier to read. The second objective is that the user can browse the web, where the application will manipulate the website in a way that will make it easier to see the web site for people with dyslexia. We decided that we will create an application for the Android operating system, but we had problems parsing the document pdf. We tried also to build on the platform PhoneGap but we also had a problem. In the end, we managed to implement our objectives for the operating system iOS. The application will be available for free on the App Store.

Keywords: dyslexia, Android, PhoneGap, iOS, mobile application

1 Uvod

Za diplomsko nalogo smo imeli priložnost izdelati aplikacijo, ki bo v pomoč osebam z bralnimi težavami. Obstaja več vrst motenj pri branju, naš izdelek je osredotočen na osebe z disleksijo. V sklopu te diplome bomo podrobneje pojasnili kaj je disleksija ter na kakšen način smo izdelali aplikacijo, da je njenim uporabnikom v pomoč pri branju. Disleksikom lahko olajšamo branje že s spremembo ozadja, povečavo pisave, uporabo prilagojene pisave, ustrezno poravnavo teksta, večjim razmikom med vrsticami ... Potrebno se je tudi zavedati, da se disleksija razlikuje od posameznika do posameznika, zato ni univerzalnega načina za izdelavo aplikacije ter njenih nastavitev. Pri izdelavi aplikacije smo sodelovali z Naravoslovnotehniško fakulteto, kjer smo skupaj določili na kakšen način izdelati aplikacijo. Odločili smo se, da izdelamo aplikacijo, ki bo uporabniku pomagala pri branju e-knjig ter pri branju spletnih strani. Pri obeh primerih smo dodali nastavitve, s katerimi si uporabnik lahko dodatno nastavi barvo ozadja, velikost pisave, barvo pisave, vrsto pisave ter jezik, v katerem se predvaja govorni vmesnik teksta (angl. Text to Speech). Naša motivacija za izgradnjo aplikacije je bila izdelati izdelek, ki bo omogočal več funkcionalnosti kot povprečna aplikacija za pomoč disleksikom. Večina aplikacij (za disleksike) omogoča samo branje e-knjig ali pa branje spletnih strani, naša aplikacija pa omogoča oboje. Poleg tega smo želeli uporabnikom ponuditi izdelek brezplačno, ker je veliko podobnih aplikacij plačljivih. Cene teh aplikacij so relativno visoke. Naša motivacija je bila izdelati realen projekt, ki bo v pomoč ljudem in bo tudi dejansko uporaben v praksi. Aplikacijo smo poizkusili izdelati na tri načine: za platformo Android, kot hibridno aplikacijo z tehnologijo PhoneGap ter za platformo iOS, vendar smo imeli popoln uspeh samo z operacijskim sistemom iOS. Na koncu smo povzeli dobre in slabe lastnosti razvoja za vsakega od treh uporabljenih pristopov. Sama aplikacija bo pred ponudbo v spletni trgovini App Store imela vsaj še eno nadgradnjo, ki bo izboljšala njeno delovanje.

V drugem poglavju smo opisali, kaj je disleksija ter na kakšen način lahko izboljšamo branje disleksikom. Poglavji tri in štiri sta namenjeni razvoju aplikacije za platformi Android in iOS ter razvoju z orodjem PhoneGap. V poglavju pet smo primerjali razvoj na posameznih platformah, kjer smo opisali pozitivne ter negativne vidike posameznih pristopov. Sklepne misli so opisane v poglavju 6.

2 Disleksija

Da lahko izdelamo aplikacijo, ki bo v pomoč disleksikom, je potrebno razumeti, kaj je disleksija ter kako lahko olajšamo branje osebam z disleksijo. Psihoterapevtka Karolina Radovanovič opiše disleksijo kot kombinacijo zmožnosti in težav, ki vplivajo na učni proces branja in pisanja [5]. Disleksija se kaže v težavah pri povezovanju posameznih črk v besede, zamenjevanju črk, pa tudi v slabši prostorski orientaciji. Disleksiji podobno stanje, pri katerem pa so težave povezane s števili in matematiko, imenujemo diskalkulija.

Disleksija in diskalkulija nista povezani z intelektualnimi sposobnostmi osebe, temveč gre za osebo, ki deluje predvsem pod vplivom desne polovice možganov, medtem ko pri večini ljudi prevladuje leva polovica možganov. Zato so disleksiki pogosto bolj ustvarjalni, umetniško nadarjeni in premorejo več empatije. Znano je namreč, da se v levi možganski polovici izvajajo bolj miselni procesi, ki temeljijo na logiki, linearnosti in zaporedju. Leva polovica je tako bolj razumsko analitična ter skrbi za govor, pisanje in branje. V desni možganski polovici pa tečejo bolj ustvarjalni procesi, osnovani na intuiciji in čustvih. Desna polovica je tako bolj ustvarjalne narave ter skrbi za neverbalne načine izražanja in razumevanja, vizualno zaznavanje, barve, glasbo, ritem ... Pri disleksikih torej prevladuje desna možganska polovica, zato so lahko zelo uspešni predvsem na področjih, kjer vlada svoboda, pa naj bo to znanost ali umetnost. Osebe z disleksijo so lahko nadpovprečno nadarjene in sposobne, kot so bili na primer znani disleksiki Leonardo da Vinci, Albert Einstein in Pablo Picasso. Po ocenah naj bi bilo disleksikov približno 30 odstotkov vse populacije [5].

Branje nadzoruje leva možganska polovica in je bolj logične narave. Čeprav je to na videz enostavna naloga, je sestavljena iz zaporedja medsebojno tesno povezanih akcij, ki zahtevajo od bralca določene sposobnosti. Pri branju besede mora tako najprej iz napisanega izluščiti posamezne črke ter jih ločiti od drugih črk podobne oblike. Nato mora črke prepoznati, zato se mora spomniti oblike posameznih črk. Sledi povezovanje črke z ustreznim glasom, kar pomeni, da mora razlikovati tudi glasove. Na koncu pa mora ustrezno zaznati še vrstni red črk, da lahko besedo tudi smiselno prebere. Če je ena ali več teh sposobnosti pri otroku manj razvita kot druge, govorimo o specifičnem primanjkljaju [5], katerega posledica je, da branje ni tako tekoče in pravilno, kot bi pričakovali glede na otrokove preostale sposobnosti.

Disleksijo in diskalkulacijo pri otrocih prepoznamo po naslednjih znakih [5]:

- Otrok počasneje bere ali piše.
- Pri branju včasih izpusti besedo ali vrstico.

- Ima težave pri prepisovanju s table v zvezek.
- Zamenjuje podobne črke b in d, n in u ali podobno zvoneče črke d in t, p in b ali simbole + in x.
- Prebere ali zapiše anagrame besed: angleški/angelski.
- Zmede se pri vrstnem redu števk: 29/92.
- Ne zmore šteti nazaj od 100 do 0.
- Ima težave pri razlikovanju med levo in desno.

Zgoraj opisani znaki nam lahko služijo tudi kot podlaga za določanje funkcionalnosti aplikacije, s katero bi zmanjšali težave disleksikov pri branju.

2.1 Priporočila za disleksike

Da lahko razvijemo aplikacijo, ki bo v pomoč disleksikom, je najprej potrebno spoznati priporočila [12], ki olajšajo branje osebam z disleksijo. Priporočil je veliko, spodaj smo povzeli najpomembnejša priporočila, po katerih smo se zgledovali ob razvoju naše aplikacije.

2.1.1 Tekst

Pri tekstu so najpomembnejši medij, na katerem se tekst nahaja, pisava, s katero je tekst zapisan, ter postavitev strani.

Medij:

- Papir, na katerega pišemo, naj bo dovolj debel, da se skozenj ne vidi, kar je napisano na drugi strani lista.
- Papir ne sme biti svetleč.

- Potrebno se je izogibati beli barvi za ozadje, ker je moteča. Namesto tega lahko uporabimo krem ali pastelne barve.

Pisava:

- Uporablja naj se osnovna pisava z enakovrednim razmikom med znaki. Pisava naj bo brezserifna, kot sta primer: Arial ali Comic Sans [6].
- Velikost pisave naj bo od 12 do 14 t.e. (tipografskih enot), nekateri disleksiki pa uporabljajo tudi večjo pisavo.
- Uporablja naj se temen tekst na svetli podlagi, vendar ne popolnoma beli.
- Izogibamo se zeleni, rdeči in roza barvi, ker so težko razločljiva za barvno slepe osebe.
- Izogibamo se poševni in podčrtani pisavi, ker disleksikom izgleda, kot da je tekst skupaj, brez presledkov.
- Izogibamo se tekstu napisanim s samimi velikimi črkami.

Postavitev strani:

- Poravnava naj bo leva.
- Vrstica naj ne bo predolga, največ 60 do 70 znakov.
- Razmik med vrsticami naj bo 1,5 vrstice.

2.1.2 Govorni vmesnik

Če želimo, da tekst predvaja govorni vmesnik (angl. Text to Speech), moramo upoštevati naslednje:

- Poslušanje dokumenta traja dlje časa kot branje teksta.
- Po predvajanju naslova naj se govorni vmesnik ustavi in šele nato nadaljuje.
- Izogibamo se rimskim številkam.

- Nekateri govorni vmesniki imajo probleme z branjem tabel.
- Izogibamo se tekstu, ki je v slikah.
- Branje večjih števil vzame dlje časa.

2.1.3 Spletno načrtovanje

Za načrtovanje spletne strani, ki je primerna za disleksike, je potrebno upoštevati naslednje:

- Navigacija na spletni strani mora biti enostavna.
- Za razporeditev teksta se uporablja grafične elemente in slike.
- Večje število grafičnih elementov povzroči, da je tekst težje berljiv.
- Lahko se uporabi alternativna predstavitev spletne strani.
- Premikajoč/animiran tekst lahko povzroča probleme pri branju.
- Za večino uporabnikov je najboljša možnost, da je tekst temnejše barve, ozadje pa blede.
- Spletna stran ima lahko možnost izbire barve ozadja.
- Izogibamo se zeleni, rdeči in roza barvi, ker so težko razločljive za osebe z barvno slepoto.
- Priporočljivo je, da si uporabnik lahko sam nastavi pisavo, velikost pisave in barvo ozadja.

2.2 Pisava

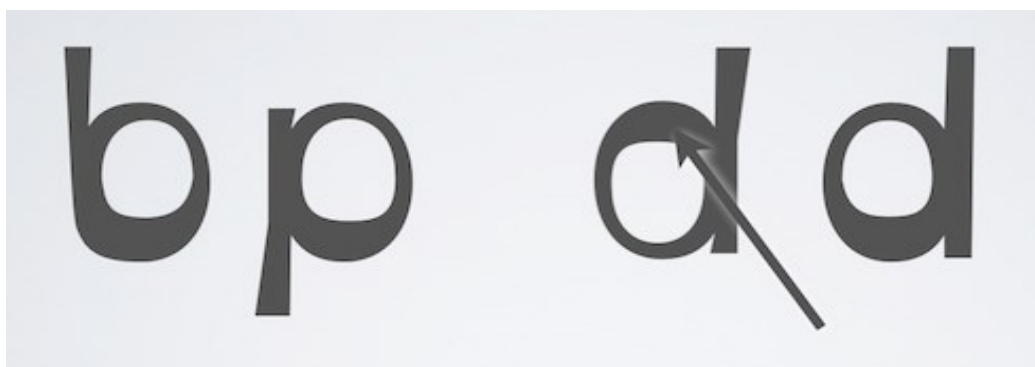
Za pomoč osebam z disleksijo je na voljo več vrst pisav. Nekatere od teh pisav so plačljive, nekatere pa ne. Za našo aplikacijo smo uporabili odprtokodno pisavo OpenDyslexic [2]. Pisava OpenDyslexic (slika 1) je novejša odprtokodna pisava namenjena lažjemu branju za

osebe z disleksijo. Sama pisava vsebuje navadno, odebeljeno, poševno ter odebeljeno poševno pisavo. Pisava se neprestano nadgrajuje in izboljšuje glede na povratne informacije oseb z disleksijo. Pisava ni plačljiva, vendar avtor vseeno prosi za referenco na spletno stran [2], kjer si je možno prenesti pisavo.

Gill Sans	rn m	MW	dpqb	l l i j j
Verdana	rn m	MW	dpqb	l1IijJ
OpenDyslexic	rn m	MW	dpqb	l1liijJ
Times	rn m	MW	dpqb	l1IijJ
Helvetica	rn m	MW	dpqb	l1liijJ

Slika 1: Prikaz pisave OpenDyslexic.

Pisava OpenDyslexic je bila ustvarjena, da olajša branje osebam z disleksijo. Posamezni znaki te pisave so spodaj odebeljeni (slika 2). Na ta način se prepozna smer znaka oz. je razvidno ali je znak obrnjen na glavo ali ne. Ne smemo pozabiti, da disleksiki tudi rotirajo znake. Obteženi znaki tudi pomagajo lažje zaznati posamezno vrstico v tekstu. Unikatna oblika znakov prepreči zmedo glede rotiranja in menjave znakov. OpenDyslexic ima tudi širši razmik med znaki ter unikatni poševni stil.



Slika 2: Prikaz odebelitve na pisavi OpenDyslexic. Znak s puščico je napačno obrnjen, kar je razvidno iz odebelitve na znaku.

Vendar pisava OpenDyslexic ni priporočljiva za vse. Nekateri uporabniki lažje berejo z pisavami, kot so: Comic Sans [6], Gill Dyslexic [8] ali Verdana ... Primer pisave Comic Sans in Gill Dyslexic vidimo na Sliki 3 ter Sliki 4. Za pisavo OpenDyslexic še ni uradne raziskave. Pisava je bila ustanovljena na manjši skupini oseb z disleksijo, nato pa se je nadgrajevala glede na povratne informacije uporabnikov. Pisavo lahko namestimo na vrsto naprav in platform kot so: Windows, Mac, iOS (iPhone, iPad), Linux.



Slika 3: Primer pisave Comic Sans.

Gill Dyslexic is a font designed to make words easier to read by people with dyslexia. What do you think?

Slika 4: Primer pisave Gill Dyslexic.

3 Razvoj aplikacije

Po raziskavi trga za Google Play Store ter za Apple App Store smo prišli do sklepa, da ima Google Play Store zelo malo aplikacij, ki so namenjene osebam z disleksijo. Te aplikacije, ki pa obstajajo, so več ali manj namenjene izobraževanju o disleksiji ter niso namenjene kot pripomoček za branje. Najdemo tudi nekaj izobraževalnih iger za otroke, vendar te še vedno ne služijo kot bralne aplikacije. Nasprotno pa ima Apple App Store veliko večje število aplikacij, ki so namenjene branju e-knjig za disleksike ter branju spletnih strani. Večina aplikacij za disleksike je plačljivih, vendar najdemo tudi aplikacije, ki niso plačljive. Večinoma se te aplikacije delijo na aplikacije za branje e-knjig ali aplikacije za branje spletnih strani. Kar zadeva cene aplikacij je razpon od brezplačnih do aplikacij, ki imajo ceno preko 60 evrov.

Po teh raziskavah smo se odločili, da naredimo aplikacijo za Google Play Store, saj ima manjše število uporabnih aplikacij. Kasneje smo naleteli na težave pri razvoju za operacijski sistem Android, kar je opisano v nadaljevanju. Poizkusili smo razviti aplikacijo tudi z uporabo spletnih tehnologij in orodjem PhoneGap, vendar ima tudi to orodje svoje omejitve.

Med raziskavo trga obstoječih aplikacij nismo našli aplikacije, ki bi omogočala branje e-knjig ter spletnih strani. Zato smo se odločili, da bo naša aplikacija omogočala oboje, hkrati pa smo želeli, da bi uporabnik imel dodatne nastavitve, ki bi omogočale prilagoditev pri branju e-knjig ter spletnih strani. Pri branju e-knjig smo želeli dodati nastavitve, ki omogočajo:

- Spremembo barve ozadja.
- Spremembo barve pisave.
- Nastavitev velikosti pisave.
- Nastavitev razmika med vrsticami.
- Govorni vmesnik, ki bo imel možnost nastavitve jezika, v katerem se predvaja tekst, spremembe hitrosti predvajanja ter nastavitve glasnosti predvajanja.
- Nastavitev, ki povrne vse nastavitve na privzeto vrednost (angl. default value), če se uporabnik zaplete z nastavitvami.

Pri branju spletnih strani smo želeli dodati spletne nastavitve, kjer si uporabnik lahko nastavi:

- Vrsto pisave, kjer ima na voljo privzeto pisavo spletne strani ali pa pisavo OpenDyslexic.
- Velikost pisave na spletni strani.
- Barvo ozadja, kjer ima možnost nastaviti ozadje, ki je najprimernejše za uporabnika.
- Nastavitev, ki nam povrne spletne nastavitve na privzete vrednosti.

Pri vseh nastavitvah smo se zgledovali po priporočilih, kako olajšati disleksikom branje e-knjig ter spletnih strani. Pri razvoju za platformo Android in z orodjem PhoneGap nam ni uspelo doseči vseh ciljev, smo pa bili uspešni pri platformi iOS.

Sledi opis poskusa razvoja za sistem Android in z orodjem PhoneGap ter kratek opis razvojnega okolja za platformo iOS, za katero smo na koncu aplikacijo tudi realizirali.

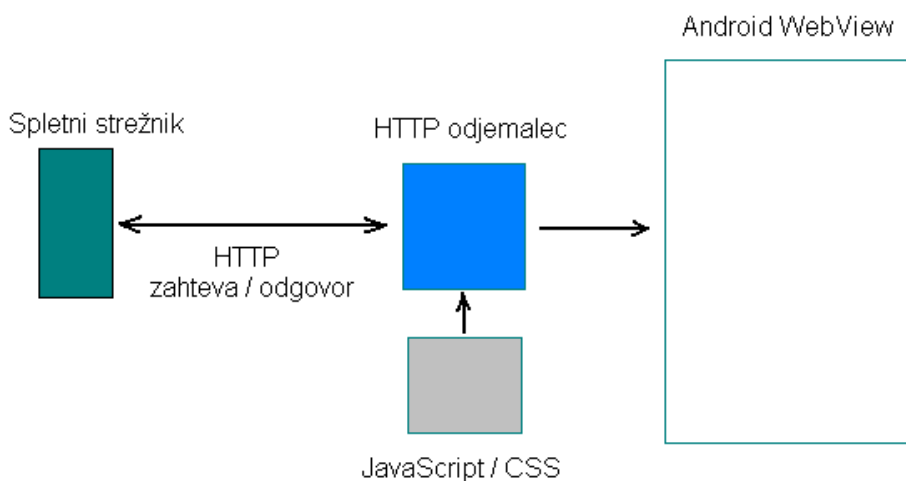
3.1 Android

Pri razvoju za Android smo naleteli na dve težavi, ki sta onemogočili nadaljnji razvoj za to platformo. Prva je težava pri razčlenjevanju formata pdf, druga pa težava pri prikazovanju modificirane (za disleksike) spletne strani.

Velika večina e-knjig je v formatu pdf, zato smo tudi potrebovali razčlenjevalnik pdf. Android ima nekaj načinov za prikaz pdf dokumenta, vendar ne omogoča razčlenjevanja dokumenta. Na spletu obstaja nekaj spletnih strani, kjer lahko kupimo razčlenjevalnik pdf, vendar je cena previsoka. Iskali smo predvsem brezplačne odprtokodne rešitve, vendar jih za sistem Android nismo našli.

Druga težava je bila pri prirejanju spletne strani na način, da je ustrežnejša za branje uporabnika z disleksijo. Po nasvetih s spletnega portala »Stack Overflow« [10] je najprimernejši način za doseg tega, da s pomočjo odjemalca pošljemo zahtevo HTTP na

spletni strežnik, nato pa v to preneseno spletno stran dodamo še slogovno predlogo (angl. Cascading Style Sheet), s katero modificiramo spletno stran (slika 5).



Slika 5: Zahteva HTTP z dodajanjem slogovne predloge (CSS).

Ta pristop se je izkazal za neuspešnega, saj ni prikazalo vseh slik spletne strani, prav tako pa hiperpovezave na spletni strani niso delovale pravilno. Zaradi teh težav, ki smo jih imeli, je tudi postalo razumljivo, zakaj ima Google Play Store tako malo bralnih aplikacij za disleksike. To je tudi bil razlog, da smo se odločili, da poizkusimo z orodjem PhoneGap.

3.2 PhoneGap

Android in iOS sta vsekakor v ospredju, vendar tudi Windows Phone lahko marsikaj doseže. Pri razvoju aplikacij za več operacijskih sistemov nam pomagajo spletne tehnologije in orodja, kot je PhoneGap [7].

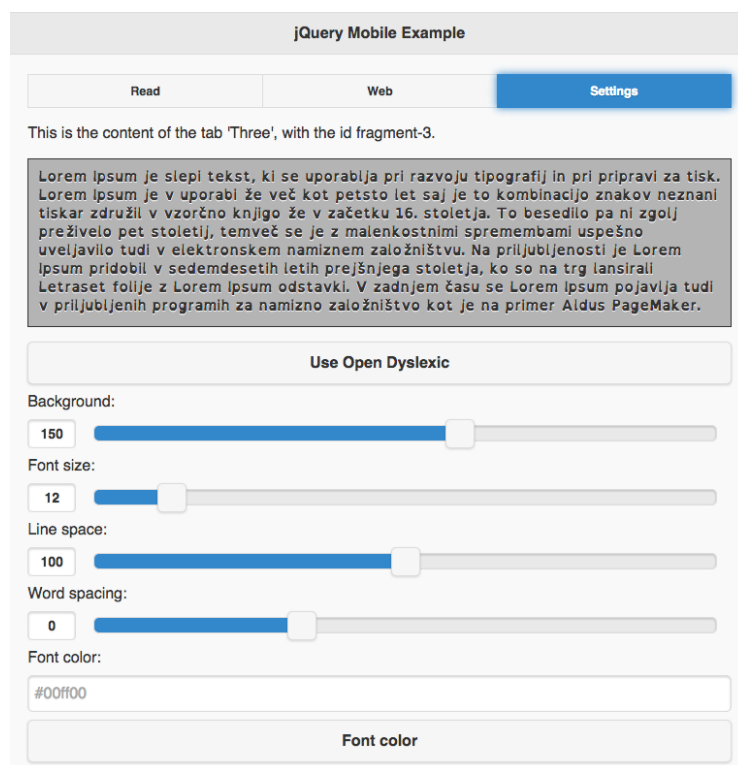
Če želimo s svojo mobilno aplikacijo pokriti kar se da veliko število uporabnikov pametnih telefonov, je potrebno narediti različico aplikacije vsaj za Android in iOS. To je časovno in finančno potratno, zato vse več razvijalcev išče rešitve, ki omogočajo enostaven razvoj za več platform hkrati. Pri podpori za več platform pri razvoju aplikacij uporabimo spletne tehnologije, kot so HTML, CSS in JavaScript (slika 6).



Slika 6: Osnove tehnologije za PhoneGap.

Orodje PhoneGap [7] je odprtokodna rešitev, ki nam omogoča, da z uporabo sodobnih spletnih tehnologij razvijemo mobilne aplikacije za vse pomembnejše mobilne operacijske sisteme: iOS, Android, BlackBerry, WebOS, Windows Phone 7, Symbian in Bada. Kar nas lahko razveseli je tudi to, da je PhoneGap odprtokodno in brezplačno orodje. Glavni gradniki orodja PhoneGap so HTML5, CSS3 ter JavaScript, lahko pa nam je v pomoč tudi knjižnica jQuery ter jQuery Mobile. jQuery olajša delo s programskim jezikom JavaScript, jQuery Mobile pa nam pomaga z izgledom aplikacije ter z dogodki (angl. events) v aplikaciji.

S pomočjo spletnih tehnologij smo želeli izdelati bralno aplikacijo (slika 7), ki bi delovala na platformah iOS in Android, in na prvi pogled je kazalo, da bomo tako aplikacijo lahko tudi uspešno izdelali. Vendar pa je potrebno pogledati tudi negativno stran orodja PhoneGap. Za začetek je potrebno povedati, da aplikacija, izdelana z orodjem PhoneGap, deluje na mobilni napravi preko spletnega vmesnika WebView, zato je na splošno počasnejša. Ima počasnejše prehode med scenami, počasnejše pa je tudi samo delovanje z vmesniki telefona, kot so [16] dostop do geolokacije, dostop do kamere, dostop do kontaktov ali dostop do datotečnega sistema.



Slika 7: Aplikacija izdelana z orodjem PhoneGap.

Ker se pri taki rešitvi pretežno uporablja programski jezik JavaScript, smo tukaj lahko uporabili knjižnico za razčlenjevanje pdf formata. To knjižnico je razvilo podjetje Mozilla, imenuje se PDF.js [1]. S pomočjo te tehnologije sicer ni bilo težav pri razčlenjevanju pdf formata, se je pa pojavil nov problem.

Eden od ciljev, ki smo si ga zastavili, je tudi ta, da bo uporabnik imel možnost brati spletne strani s pomočjo posebne pisave, ki je prilagojena osebam z disleksijo. Orodje PhoneGap sicer ima vgrajen svoj spletni brskalnik, imenovan InAppBrowser, vendar v ta brskalnik ni bilo mogoče vgraditi posebne pisave za disleksike. Težava je bila v tem, da smo pisavo OpenDislexic lahko vgradili le v samo aplikacijo, ne pa tudi v brskalnik InAppBrowser. Razlog je ta, da navedeni brskalnik deluje ločeno od aplikacije in ne prepozna pisave, ki smo jo vgradili v aplikacijo. Poleg tega se je pojavila še težava z govornim vmesnikom, saj ga samo orodje PhoneGap nima. Potrebno bi bilo napisati svoj vmesnik med orodjem PhoneGap in platformo Android / iOS.

Ker ponovno nismo mogli doseči vseh ciljev, ki smo si jih zadali, smo se odločili prekiniti razvoj s to tehnologijo in ponovno začeti projekt na platformi iOS.

3.3 iOS

iOS [3] je operacijski sistem podjetja Apple. Zasnovan je bil za mobilne telefone iPhone, kasneje pa se je začel uporabljati tudi za naprave iPad, iPod ter Apple TV. Če želimo razvijati aplikacije za ta sistem, potrebujemo znanje programskega jezika Objective-C [4] v kombinaciji s programskim jezikom C. Najpogostejše razvojno orodje je Xcode [9], ki je produkt podjetja Apple. V spodnjih poglavjih je podrobnejši opis razvojnega okolja Xcode ter programskega jezika Objective-C.

3.3.1 Razvojno okolje Xcode

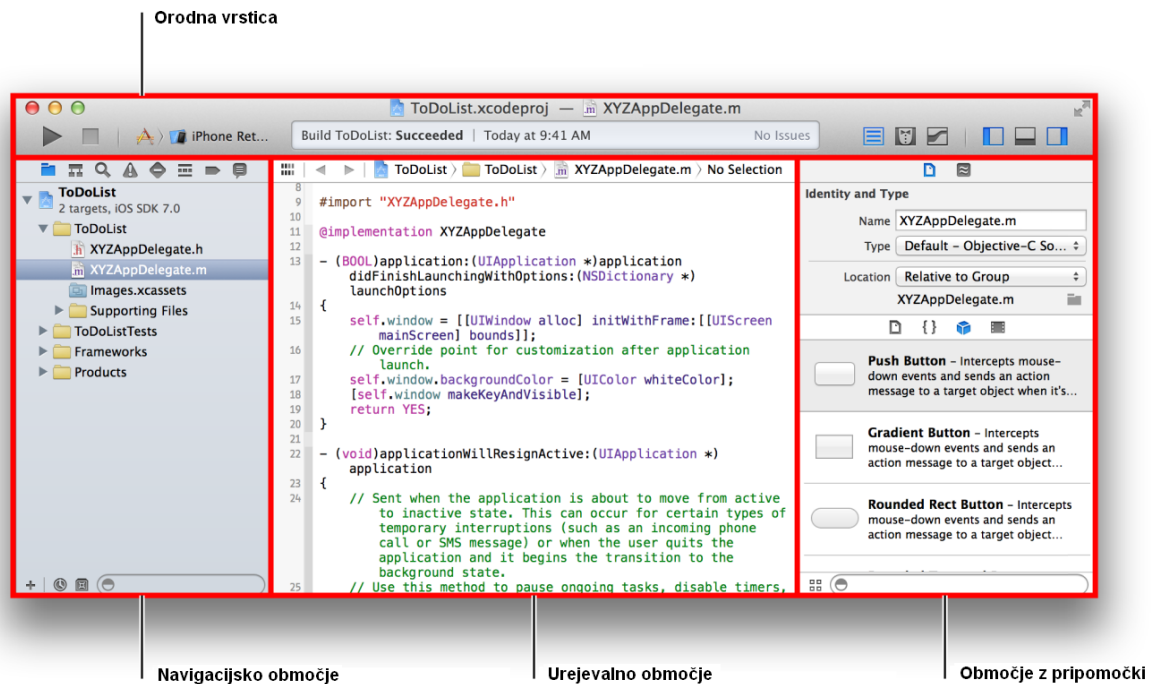
Xcode [9], [15] je razvojno okolje (IDE), namenjeno operacijskemu sistemu OS X ter iOS. Samo razvojno okolje je brezplačno, vendar če želimo objaviti aplikacijo na spletni trgovini App Store, je potrebno kupiti licenco razvijalca, ki trenutno stane 99 ameriških dolarjev. Prav tako ne moremo prenesti aplikacije na napravo (iPhone oz. iPad), ne da bi si prej ustvarili poseben certifikat, imenovan provisioning profile, za kar je potrebno uporabiti licenco razvijalca.

Samo okolje se je razvilo v sodoben produkt, ki združuje orodja za razvijanje, razhroščevanje, oblikovanje in simulacijo aplikacije. S pomočjo okolja Xcode je razvoj na platformi iOS veliko hitrejši in kvalitetnejši kot na platformi Android. Že samo oblikovanje aplikacije in izdelava navigacije v aplikaciji sta veliko hitreje izdelana. Preprosto povežemo posamezne grafične elemente v eni sami datoteki s končnico storyboard in navigacija med pogledi je končana. Nato pa še povežemo grafične elemente s kodo, ki smo jo napisali, in izdelek lahko testiramo na vgrajenem simulatorju.

Grob opis razvojnega okolja Xcode s slike 8:

- Orodna vrstica je namenjena prenosu aplikacije na simulator ali na napravo ter preklopu med posameznimi pogledi projekta.
- Navigacijsko območje je namenjeno za pregled nad datotekami, ki so vključene v projekt, obenem pa si tukaj lahko pogledamo opozorila in napake, ki jih naš projekt lahko vsebuje.
- Urejevalno območje je namenjen urejanju kode ter grafičnem oblikovanju aplikacije.

- Območje s pripomočki je namenjeno dodajanju grafičnih elementov na posamezne poglede projekta, lahko pa se tudi dodaja koda.



Slika 8: Razvojno okolje Xcode.

3.3.2 Objective-C

Če želimo razvijati za operacijski sistem iOS, potrebujemo znanje programskega jezika Objective-C [4], [17]. Objective-C je splošno namenski objektno orientiran programski jezik in je podobno kot C++ nadgradnja programskega jezika C. Ta jezik primarno uporablja podjetje Apple, namenjen pa je operacijskemu sistemu OS X in iOS. Jezik je bil razvit leta 1980, uporabljalo ga je podjetje NeXT za svoj operacijski sistem NeXTSTEP, iz katerega sta se razvila OS X ter iOS. Če se poglobimo v sam programski jezik, ugotovimo, da so vsi primitivni tipi enaki kot v programskem jeziku C, objektni del pa je oblikovan po jeziku Smalltalk.

4 Aplikacija za pomoč pri branju

Našo aplikacijo smo razdelili na dva dela. Prvi del je namenjen branju e-knjig v formatu pdf, drugi del pa je namenjen branju prilagojenih spletnih strani. Raziskali smo, kako se dokument pdf uvozi v aplikacijo, izbrati smo si morali primeren razčlenjevalnik pdf, razumeti je bilo potrebno delo z govornim vmesnikom ter najti smo morali način, kako v spletno stran uvoziti novo slogovno predlogo CSS. Vse to je opisano v spodnjih poglavjih. Aplikacijo smo razdelili na štiri zavihke. Ti zavihki so: Library, Web, Settings ter About.

Delovanje aplikacije je preprosto; če želimo uvoziti dokument pdf (e-knjigo) v aplikacijo, lahko to storimo preko programa iTunes z operacijo povleci in spusti ali pa uporabimo zunanjo aplikacijo (npr. za elektronsko pošto), ki nam uvozi dokument pdf v aplikacijo. Ko imamo dokument uvožen, ga lahko začnemo brati. V aplikacijo smo dodali tudi navodila za uporabo, ki so shranjena v formatu pdf, tako da jih lahko beremo kot e-knjigo.

Če želimo brati e-knjigo v načinu za disleksike (zavihek Library), lahko z gumbom (zgoraj desno) vklopimo poseben pogled. Ko imamo vklopljen ta pogled, imamo na voljo gumb, ki nas popelje v nastavitve za branje, kjer si uporabnik prilagodi izgled e-knjige. Na voljo imamo tudi govorni vmesnik, ki nam predvaja vsebino e-knjige.

Če uporabnik želi brskati po spletu, lahko izbere zavihek Web, kjer je vnosna vrstica, kamor vpiše spletni naslov (naslov URL). Če želi spletno stran modificirati, se zgoraj desno nahaja gumb, ki odpre spletne nastavitve, kjer lahko spremeni vrsto pisave, velikost pisave ter barvo ozadja.

Aplikacija ima še zavihek Settings, ki nas popelje na nastavitve za e-knjige (Book Settings) ter nastavitve za splet (Web Settings).

Spodaj sledi opis izgradnje aplikacije po zavihkih.

4.1 Uvoz dokumenta pdf

Pri branju knjig je bilo potrebno najprej implementirati uvoz e-knjige (datoteke pdf) v našo aplikacijo. Operacijski sistem iOS podpira dva načina za uvoz datotek [14], in sicer z uporabo iTunes in uvoz med aplikacijami, npr. preko aplikacije za e-pošto.

V naši aplikaciji smo implementirali oba načina, tako da lahko uporabnik sam izbere svoj priljubljen način uvoza e-knjige v aplikacijo.

Za uvoz z uporabo iTunes smo dodali posebno nastavitev `UIFileSharingEnabled` (v nastavitvah projekta), nato pa smo dodali kodo za dostop do tega dokumenta pdf. Ko uvozimo dokument preko sistema iTunes, se ta dokument prenese v imenik znotraj aplikacije z imenom `NSDocumentDirectory`.

Drugi način uvoza aplikacije je prenos datoteke preko neke druge aplikacije. Če želimo imeti to možnost, je potrebno v nastavitvah projekta (plist oz. property list) nastaviti vrednosti, ki jih prikazuje slika 10.

Key	Value
▼ Information Property List	(15 items)
Application supports iTunes file sh	<input checked="" type="checkbox"/>
▼ CFBundleDocumentTypes	(2 items)
▼ Item 0	(4 items)
CFBundleTypeName	PDF Document
LSHandlerRank	Alternate
CFBundleTypeRole	Viewer
▼ LSItemContentTypes	(1 item)
Item 0	com.adobe.pdf
Localization native development re	English
Bundle display name	\${PRODUCT_NAME}
Executable file	\${EXECUTABLE_NAME}

Slika 10: Nastavitve v projektu, za uvoz dokumenta pdf.

Poleg tega je bilo treba dodati kodo, da aplikacija lahko sprejme datoteko, ki jo je poslala neka druga aplikacija, in jo tudi shrani. Datoteka se shrani v imenik Inbox znotraj imenika `NSDocumentDirectory`. Po uspešnem uvozu datoteke lahko uvoženo e-knjigo opazimo v tabeli v zavihku Library.

4.2 Razčlenjevalnik pdf

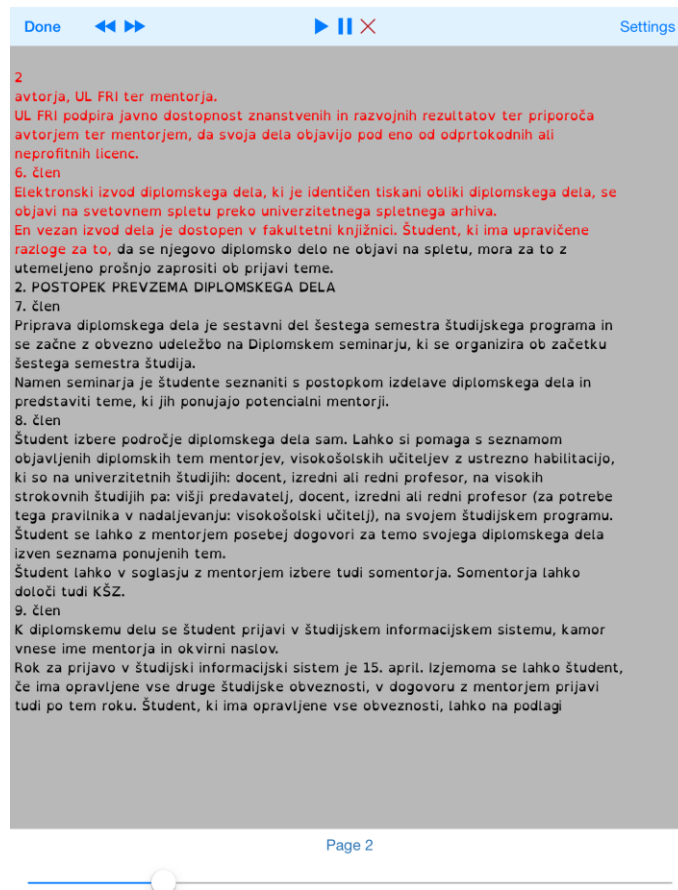
Operacijski sistem iOS ima podporo za delo z dokumenti pdf. Vendar pa je delo z njimi nizkonivojsko, kar nam je otežilo delo. Zato smo se odločili, da za ekstrakcijo teksta vzamemo že obstoječo knjižnico FastPdfKit [13].

Knjižnica FastPdfKit je v osnovi brezplačna, lahko pa je tudi plačljiva, če želimo uporabljati naprednejše funkcije. V našem primeru smo uporabili samo ekstrakcijo teksta iz dokumenta pdf, zato smo se odločili za brezplačno verzijo. Tekst je bilo potrebno izvleči iz e-knjige zato, da smo ga lahko primerno obdelali ter nad njim uporabili tudi govorni vmesnik. Knjižnica FastPdfKit nam omogoča tudi:

- prilagoditev kode za lastne potrebe,
- iskanje ključnih besed,
- razčlenjevanje,
- redno vzdrževano dokumentacijo ter,
- podpora za multimedije (video, avdio, spletne strani ...).

4.3 Govorni vmesnik

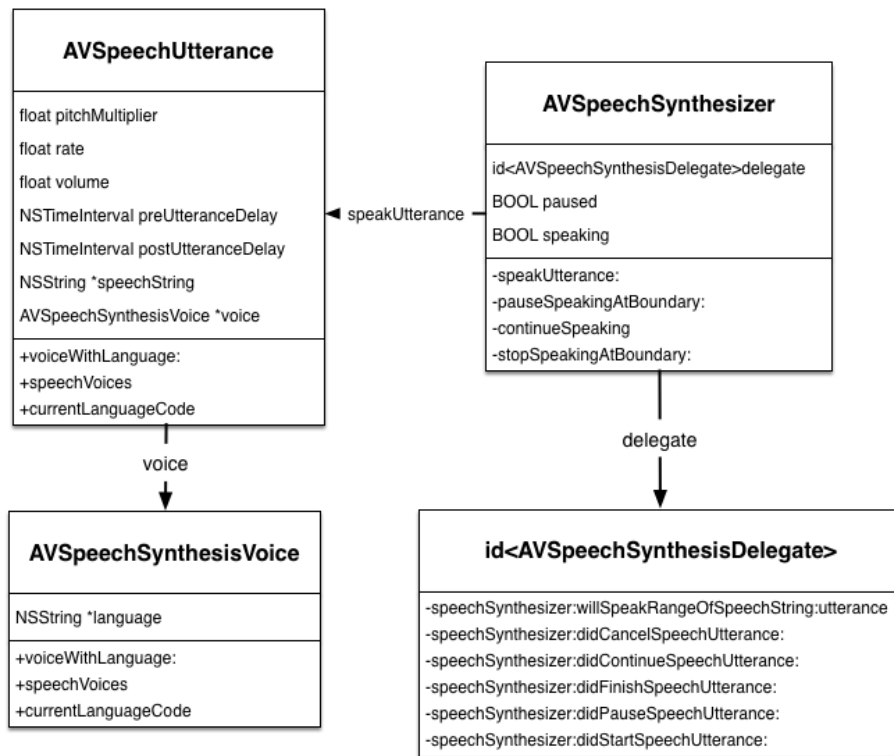
Na tem posebnem pogledu za disleksike imamo tudi možnost uporabe govornega vmesnika [11], kjer nam govorni vmesnik predvaja vsebino naše e-knjige. Govorni vmesnik vključimo s pomočjo gumba predvajaj, vgradili pa smo tudi gumb za pavzo ter gumb za stop. Aplikacija tudi označuje tekst (slika 11), ki se predvaja, tako da uporabnik lažje sledi napredku. Vgradili smo tudi možnost, da si lahko uporabnik sam nastavi jezik, v katerem želi, da se govorni vmesnik izvaja.



Slika 11: Govorni vmesnik z indikacijo o poteku govora.

Glavno vlogo pri govornem vmesniku ima objekt, ki je instanca razreda AVSpeechSynthesizer (slika 12). Temu objektu podamo kot vhodni parameter še en objekt tipa AVSpeechUtterance, ki pa vsebuje tudi znakovni niz, ki ga želimo predvajati. Objektu tipa AVSpeechUtterance lahko nastavimo:

- jakost zvoka,
- hitrost govorjenja ter,
- jezik, v katerem govori glasovni vmesnik.



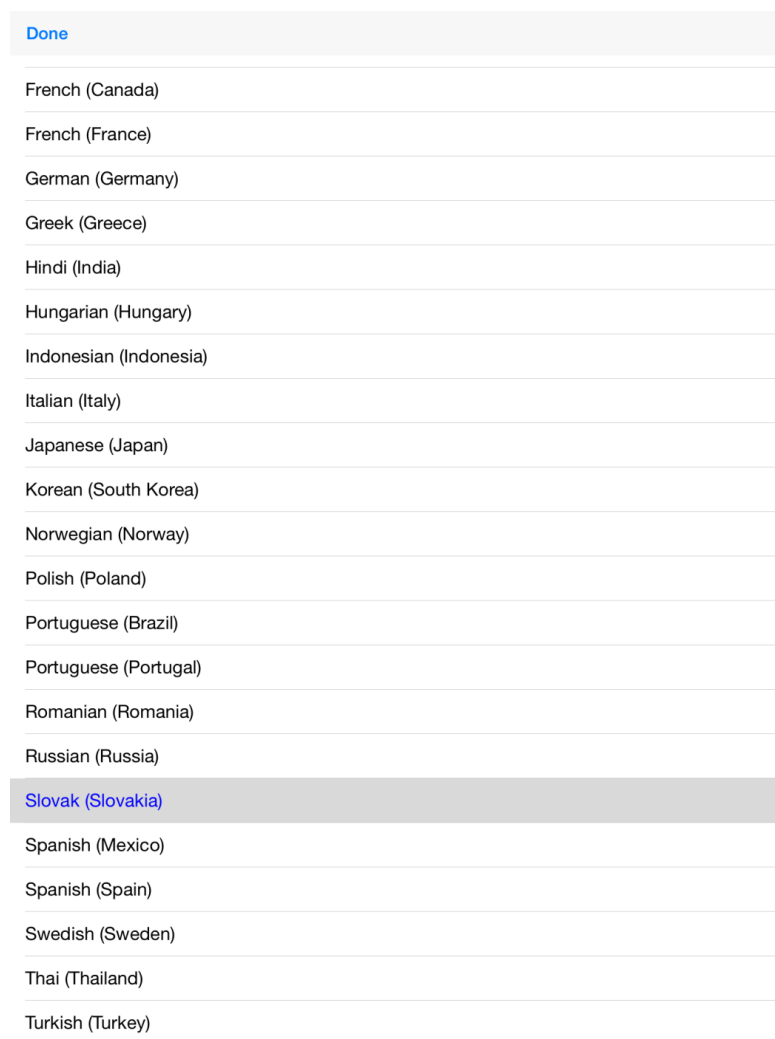
Slika 12: Komunikacija med razredi, za delovanje govornega vmesnika.

Zelo pomembne so tudi metode delegatov objekta tipa `AVSpeechSynthesizer`, s pomočjo katerih lahko pridobimo pomembne podatke o stanju govornega vmesnika, kot so:

- Ali je uporabnik prekinil izvajanje govornega vmesnika?
- Ali bo govorni vmesnik ponovno začel govoriti po prekinitvi?
- Ali je govorni vmesnik zaključil s predvajanjem?
- Ali je govorni vmesnik prešel v stanje pavze?
- Ali je govorni vmesnik začel govoriti?
- Kje se trenutno nahaja govorni vmesnik glede na podani znakovni niz?

S pomočjo teh metod delegatov imamo možnost, da ugotovimo, pri katerem znakovnem nizu se govorni vmesnik nahaja, kar pa nam omogoča barvanje teksta. Za barvanje teksta smo se odločili zato, da lahko uporabnik lažje sledi besedilu.

Pri uporabi govornega vmesnika imamo tudi možnost, da si izberemo drug jezik, v katerem se predvaja tekst. Aplikacija deluje na način, da v nastavitvah operacijskega sistema zazna katero jezikovno nastavitev imamo privzeto, nato pa nastavi govorni vmesnik na ta jezik. Še vedno pa imamo v nastavitvah možnost (slika 13), da spremenimo jezik, v katerem predvaja govorni vmesnik. Obstaja tudi možnost, da privzete jezikovne nastavitve operacijskega sistema govorni vmesnik ne podpira. V takem primeru se jezik govornega vmesnika nastavi na angleški jezik, če uporabnik uporablja slovenski jezik priporočamo nastavitev na slovaški jezik, saj je najbližji približek slovenskemu jeziku.



Slika 13: Izbira med jeziki v aplikaciji.

Primer uporabe govornega vmesnika v programskem jeziku Objective C:

```
AVSpeechSynthesizer *synthesizer = [[AVSpeechSynthesizer alloc] init];
synthesizer.delegate = self;

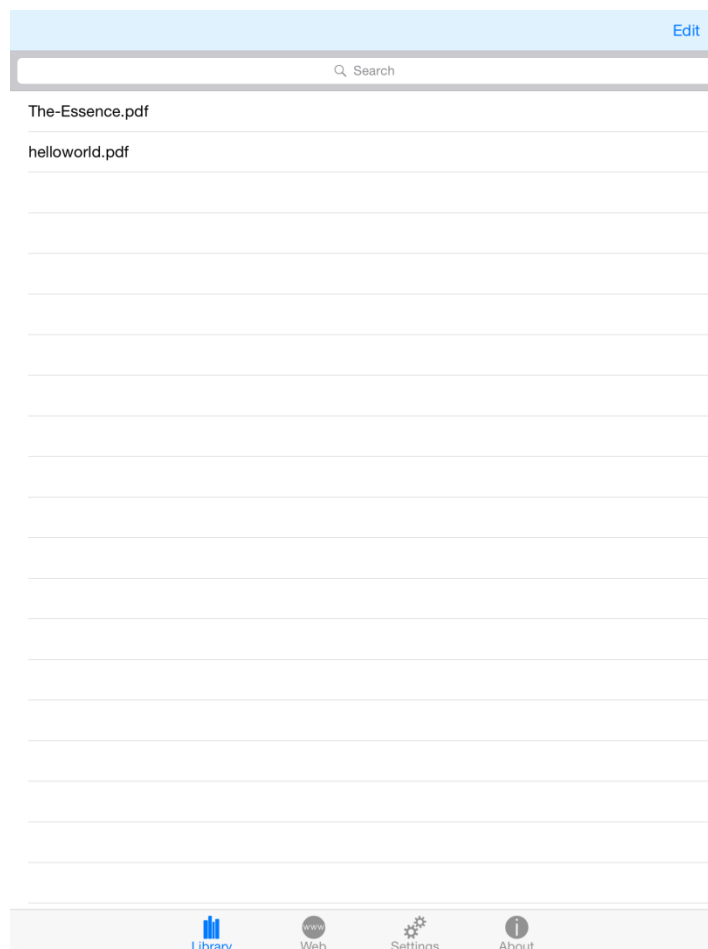
NSString *text = @"Hello world";
AVSpeechUtterance *utterance = [[AVSpeechUtterance alloc] initWithString:text];

utterance.voice = [AVSpeechSynthesisVoice voiceWithLanguage:@"en-AU"];
[synthesizer speakUtterance:utterance];
```

Koda prikazuje preprost primer, kako dosežemo, da govorni vmesnik spregovori. Začnemo s tem, da ustvarimo instanco razreda *AVSpeechSynthesizer*. Nato še ustvarimo instanco razreda *AVSpeechUtterance*, kamor kot parameter vnesemo znakovni niz (Hello world), instanci pa tudi izberemo jezik. Instanco razreda *AVSpeechUtterance* podamo instanci razreda *AVSpeechSynthesizer*, kot vhodni parameter metode in govorni vmesnik spregovori. V našem primeru nas pozdravi z »Hello world«.

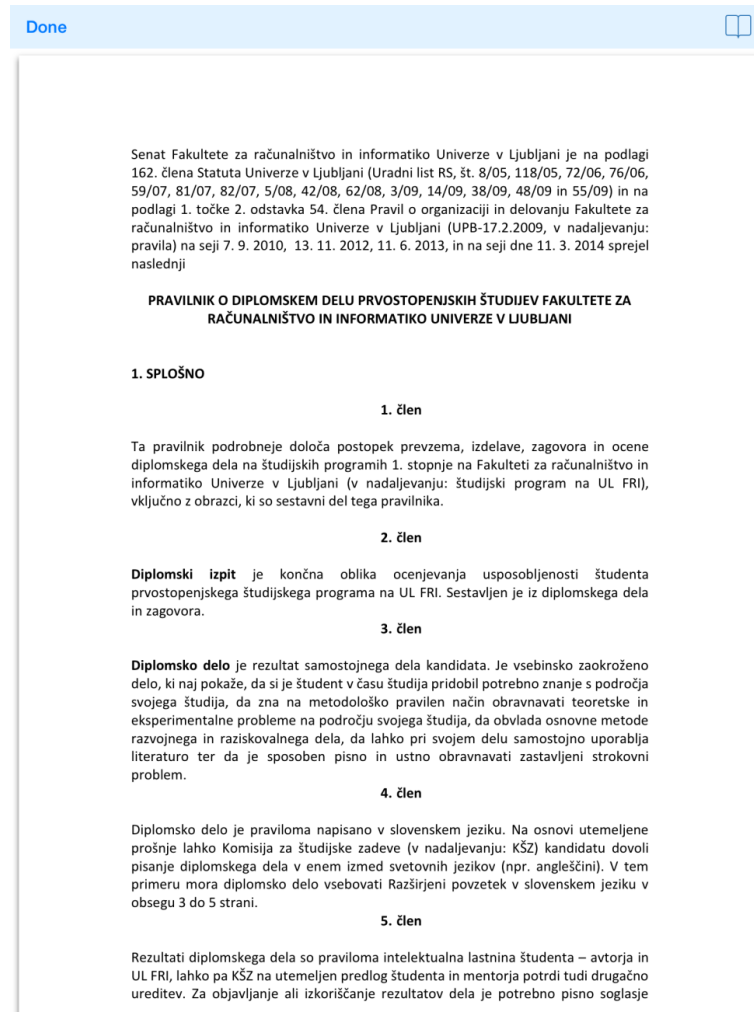
4.4 Zavihek Library

Zavihek Library (slika 14) je zavihek, kjer imamo seznam e-knjig, ki si jih uporabnik lahko izbere za branje. To se izvede tako, da se najprej ustvari objekt razreda *UITableViewController*, ki je glavni razred za poglede s tabelami. Nato aplikacija iz svojega lokalnega imenika prebere shranjene datoteke (e-knjige) in jih prikaže v tabeli. Nad tabelo ima aplikacija tudi iskalno vrstico (razred *UISearchBar*), ki nam omogoča iskanje po tabeli tako, da uporabnik lahko najde ustrezno e-knjigo, če je seznam preobsežen.



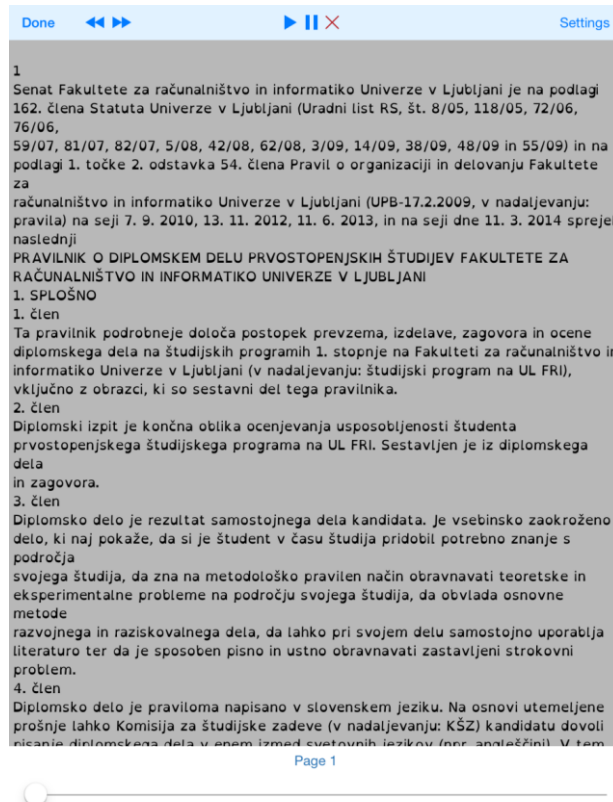
Slika 14: Zavihek Library, kjer izberemo poljubno e-knjigo.

Ob izbiri ustrezne e-knjige v zavihku Library se nam v novem objektu tipa `UIViewController` odpre izbrani dokument pdf (slika 15). Ta dokument pdf je v svoji osnovni obliki, brez kakršnihkoli sprememb. iOS uporablja za prikaz dokumenta pdf poseben spletni pogled (vmesnik `UIWebView`), ki je primarno namenjen za prikaz spletnih strani, lahko pa prikazuje tudi dokumente pdf. V vmesnik `UIWebView` smo dodali možnost povečave, tako da uporabnik lahko uporabi gesto »pinch« in pogled se mu poveča.



Slika 15: E-knjiga, prikazana z vmesnikom UIWebView.

Če želimo brati e-knjigo pod posebnimi nastavitvami za disleksike, to vklopimo z gumbom (zgoraj desno), ki nam vstavi nov objekt tipa `UIViewController` s pogledom za disleksike (slika 16). Tukaj imamo pogled, kakršnega smo si nastavili v nastavitvah za branje e-knjig. Lahko imamo drugačno barvo ozadja, posebne prilagojene črke ter spremenjeno velikost črk. Vse to lahko nastavimo v nastavitvah za branje e-knjig (Book Settings).



Slika 16: Pogled za osebe z disleksijo.

4.4.1 Nastavitve za branje

V pogledu za disleksiike imamo zgoraj desno gumb, ki nam odpre nastavitve za branje e-knjig (Book Settings), ki jih prikazuje slika 17. Nastavitve za branje omogočajo uporabniku, da si lahko nastavi oz. izbere:

- barvo ozadja,
- barvo pisave,
- velikost pisave,
- razmik med vrsticami,
- hitrost govornega vmesnika,
- jakost govornega vmesnika,
- jezik govornega vmesnika,
- privzete nastavitve.

Done

Example text:

Lorem ipsum dolor sit er elit lamet, consectetur cillum adipiscing pecu, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Nam liber te conscient to factor tum poen legum odioque civiuda.

Background color:

Font color:

Font size:

Line spacing:

Text to speech (TTS)

Speed: min max

Volume: min max

Language: English (United States)

Select language

Restore default settings:

Restore

Slika 17: Nastavitve za branje e-knjig.

Nad nastavitvami smo vstavili tekstovno polje, ki dinamično prikazuje izbrane nastavitve in njihove spremembe. V primeru, da uporabnik želi spreminjati nastavitve, lahko spremembe sproti spremlja v tekstovnem polju nad nastavitvami. Pri teh nastavitvah smo se zgledovali po priporočilih [12], kako olajšati branje disleksikom. Opis posameznih nastavitev:

- Background color: uporabnik spreminja barvo ozadja. Tukaj smo se odločili, da ne dodamo barvne palete, kjer bi si lahko uporabnik izbral poljubno barvo za ozadje, ampak smo se odločili za sivo barvo in drsnik, s katerim lahko uporabnik spreminja svetlost ozadja. Za disleksi je namreč najbolj primerno sivinsko ozadje oz. temnejša bela barva, tako da drsnik omogoča spremembo barv od črne (#000000) do bele (#FFFFFF), vedno pa se spremeni vrednost vsem trem komponentam barve: rdeči, zeleni in modri.
- Font color: podobno kot pri barvi ozadja tukaj preko drsnika spreminjamo barvo pisave.

- Font size: tukaj ima uporabnik možnost povečati ali zmanjšati velikost pisave. Namesto drsnika smo tukaj uporabili element UIStepper, ki nam omogoča, da postopno (koračno) spreminjamo velikost pisave.
- Line spacing: je razmik med vrsticami. To je ena od nastavitev, ki disleksikom omogoča lažje branje teksta. V našem primeru smo uporabili element UIStepper za nastavitve razmika med vrsticami.
- Text to speech (TTS): ima več nastavitev, ki pomagajo, da si uporabnik optimizira govorni vmesnik. Nastavi lahko hitrost predvajanja teksta, glasnost predvajanja teksta ter izbiro jezika, ki ga uporablja govorni vmesnik.
- Restore default settings: je nastavev, ki nastavi vse nastavitve na privzete vrednosti. Ta nastavev se uporablja, če se uporabnik zaplete z nastavitvami.

4.5 Zavihek Web

Zavihek Web omogoča uporabnikom, da si lahko ogledajo spletne strani preko vgrajenega brskalnika s pomočjo vmesnika UIWebView. Hkrati pa aplikacija tudi dopušča, da si uporabnik prilagodi prikaz spletne strani. Osebe z disleksijo težje berejo tekst na spletnih straneh, zato smo se odločili, da uporabniku ponudimo možnost, da si v spletnih nastavitvah (Web Settings) sam izbere možnost, na kakšen način želi brati spletno stran. To, da lahko uporabnik bere spletne strani, smo izvedli na naslednji način:

1. Uporabnik v url vrstico vpiše spletni naslov.
2. Izvede se http zahteva.
3. Če je zahteva bila izvedena pravilno, se preko vmesnika UIWebView sproži metoda delegata, ki nam pove, da je http zahteva zaključena.
4. V tej metodi s pomočjo jezika JavaScript vrnemo našo slogovno predlogo (CSS) v naloženo spletno stran.
5. Uporabniku se prikaže prirejena (modificirana) spletna stran.

Zgoraj so navedene glavni koraki algoritma za prirejanje spletnih strani, sama implementacija pa ni tako enostavna. Najprej smo v datoteko storyboard postavili grafični element

UIViewController in ga povezali z razredom UIViewController. Nato smo dodali še spletni vmesnik UIWebView, ki nam omogoča prikazovanje spletne strani, ta vmesnik pa smo še povezali z gumbi. Ti gumbi so:

- gumb za osvežitev spletne strani (ang. refresh),
- gumb za povezavo nazaj (angl. backwards),
- gumb za povezavo naprej (angl. forward),
- gumb za izvedbo (angl. Go) povezave na spletno stran glede na naslov url vrstice, kjer uporabnik vpiše spletni naslov.

Nato smo dodali še kodo, ki je povezovala funkcionalnost, tako da bi v tem trenutku že lahko uporabljali vgrajeni spletni brskalnik. Vendar je bilo potrebno še dodati spletne nastavitve, ki bi uporabniku omogočale ogled spletnih strani z nastavitvami za disleksike (Web Settings). Zato smo dodali dodaten gumb, ki nam prikaže spletne nastavitve. Delovanje aplikacije je naslednje: ko se nam spletna stran naloži v vmesnik UIWebView, se nam sproži metoda delegata webViewDidFinishLoad, kamor s pomočjo jezika JavaScript vstavimo slogovno predlogo CSS. S pomočjo tega stila priredimo spletno stran. Slogovna predloga se sestavi preko spletnih nastavitev.

Slike 18, 19 in 20 prikazujejo spletne strani pod različnimi nastavitvami. Slika 18 prikazuje originalno spletno stran brez prirejanja, slika 19 pa prikazuje isto spletno stran, na kateri je aplikacija privzeto pisavo zamenjala s pisavo za disleksike OpenDyslexic. Slika 20 prikaže spletno stran z modificiranim ozadjem, ki je primerno za disleksike, povečano pisavo ter uporabljeno pisavo OpenDyslexic.



Slika 18: Osnovni pogled, brez uporabe nastavitev.



Slika 19: Spletna stran z dodano pisavo OpenDyslexic.

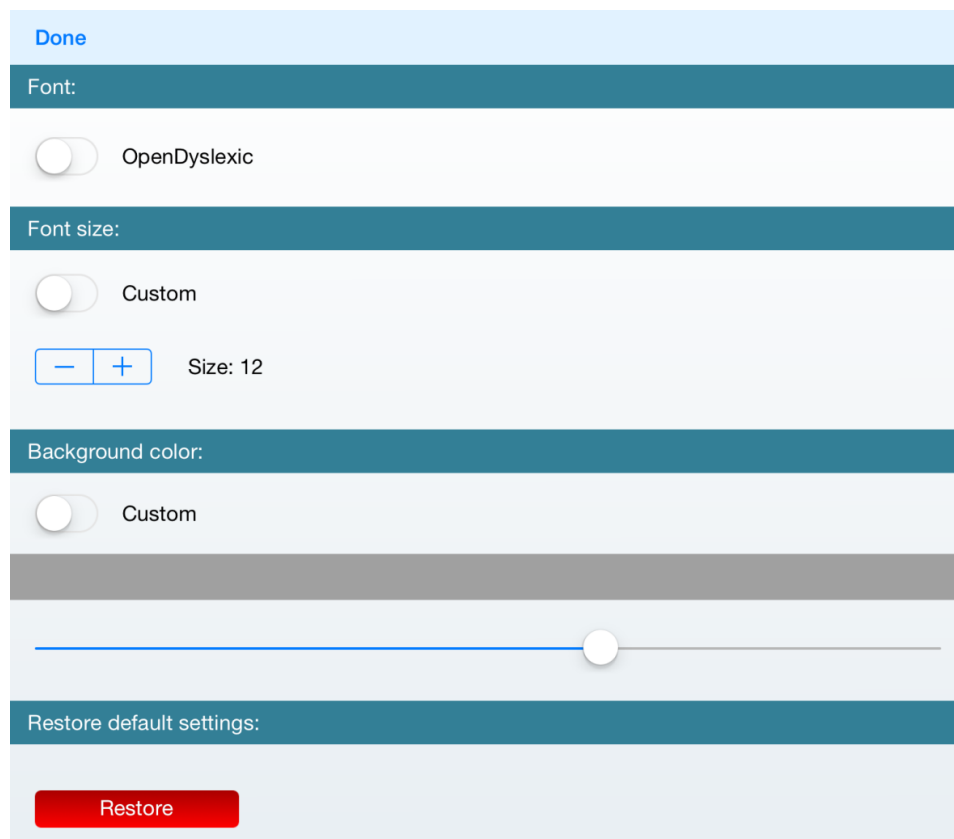


Slika 20: Spletna stran z dodano pisavo OpenDyslexic, spremenjenim ozadjem ter povečano pisavo.

4.5.1 Spletne nastavitve

Spletne nastavitve oz. Web Settings (slika 21) so namenjene uporabniku, da ima možnost spremembe izgleda spletne strani. Tako si lahko poljubno nastavi barvo ozadja, velikost pisave in vrsto pisave. Nastavitve se ob spremembi tudi shranijo, tako da jih ni potrebno ponovno nastaviti. Spodaj so opisane posamezne nastavitve:

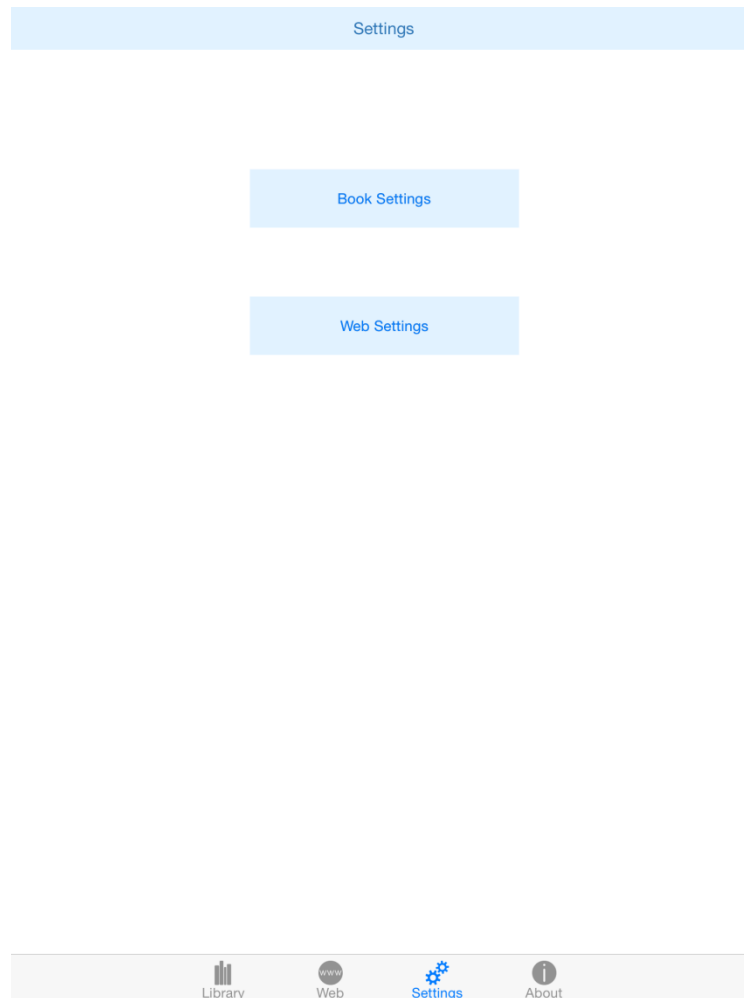
- Font: tukaj si uporabnik lahko izbere, kakšno pisavo želi uporabljati na spletni strani. Ima možnost uporabiti privzeto pisavo, ki je na sami spletni strani, ter ima možnost, da se spletna stran prikaže s posebno pisavo OpenDyslexic, ki je bila izdelana za disleksike.
- Font size: tukaj si uporabnik izbere velikost pisave.
- Background color: nastavitvev omogoča spremembo barve ozadja.
- Restore default settings: nastavi nastavitve na privzete vrednosti.



Slika 21: Spletne nastavitve.

4.6 Zavihek Settings

Zavihek Settings (slika 22) ima samo dva gumba, ki nas popeljeta na nastavitve za branje e-knjig (Book Settings) ter na nastavitve za prikaz spletne strani (Web Settings). Izvedba je preprosta, na pogled (angl. view) za nastavitve smo dodali dva gumba, ki sta povezana na ustrezne nastavitve (Book Settings in Web Settings) z elementom Segue. Ta zavihek je namenjen temu, da uporabniku olajša dostop do nastavitev, vsekakor pa imata zavihka za branje e-knjig ter za ogled spletnih strani ob sebi gumb za nastavitve, tako da so vedno na voljo uporabniku.



Slika 22: Zavihek Settings v aplikaciji.

4.7 Zavihek About

Zavihek About je namenjen predvsem predstavitvi aplikacije ter navedbi referenc odprtokodnih rešitev, ki smo jih uporabili v aplikaciji. Potrebno je dodati referenco za pisavo OpenDyslexic ter razčlenjevalnik pdf FastPdfKit. Oba izdelka sta brezplačna, vendar je potrebno imeti referenco na spletne strani avtorjev.

5 Primerjava razvitih aplikacij

V tem poglavju je povzeto delo z vsemi platformami, ki so bile uporabljene v tej diplomski nalogi. Opisane so pozitivne strani ter negativne. Pogosto se razvijalci sprašujejo, ali razviti aplikacijo na platformi PhoneGap ali razviti aplikacijo v svojem domorodnem (angl. native) okolju. Vedno najdemo razloge za in proti. S tehnologijo PhoneGap imamo možnost zgraditi aplikacijo samo enkrat in jo lahko prevedemo za več sistemov, kar je časovno ugodno. Ne smemo tudi pozabiti, da je veliko več spletnih razvijalcev kot pa razvijalcev za npr. Android, iOS, Blackberry ... To omogoča, da spletni razvijalci na hiter način izdelajo aplikacijo, brez učenja novih tehnologij.

Orodje PhoneGap ima tudi veliko negativnih učinkov v primerjavi z aplikacijami v domorodnem načinu. Je predvsem počasnejši in ne omogoča vseh funkcionalnosti, ki jih omogočajo domorodni sistemi. Spodnja tabela (tabela 1) primerjav se predvsem nanaša na našo aplikacijo in je ne smemo povzeti na splošno.

	PhoneGap	Android	iOS
Tranzicije med scenami	Počasna	Hitra	Hitra
Hitrost delovanja aplikacije	Na splošno so hibridne aplikacije počasnejše od domorodnih.	Hitra	Hitra
Razčlenjevalnik pdf	Ker se uporablja programski jezik JavaScript lahko uporabimo brezplačno knjižnico PDF.js [1].	Android nima brezplačnega razčlenjevalnika pdf. Lahko si pomagamo z razčlenjevalniki, ki so izdelani v drugih programskih jezikih, vendar je to opravilo zahtevno in časovno potratno.	iOS ima vgrajeno podporo za format pdf. Ker je ta podpora nizkonivojska, smo uporabili knjižnico FastPdfKit [13].
Hitrost razvoja	Omogoča, da v preprostem programskem jeziku JavaScript izdelamo aplikacijo ter jo prevedemo na številne platforme: Android, iOS, Blackberry ...	V našem primeru je izdelava aplikacije na sistemu Android bila najpočasnejša. Android v primerjavi z iOS nima potrebnih knjižnic ter metod za izgradnjo kvalitetne bralne aplikacije za disleksike.	Razvoj je bil hiter, vendar počasnejši kot na sistemu PhoneGap.

Vgrajen spletni brskalnik (vmesnik)	InAppBrowser	WebView	UIWebView
Dodajanje pisave (OpenDyslexic) v spletni brskalnik	Po več dneh truda nam v ta spletni brskalnik ni uspelo vstaviti pisave za disleksike. Brskalnik ima svoj obseg (ang. scope) in ne prepozna pisave, ki smo jo vgradili v aplikacijo.	S pomočjo slogovnih predlog (CSS) nam je uspelo vstaviti pisavo za disleksike v brskalnik, vendar smo imeli težave. Spletna stran ni bila pravilno prikazana, pa tudi hiperpovezave niso delovale pravilno.	S številnimi metodami, ki jih nudi razred UIWebView, tukaj nismo imeli težav.
Programski jezik	JavaScript	Java	Objective-C
Stilsko oblikovanje aplikacije	S pomočjo slogovnih predlog CSS.	Počasno oblikovanje, vsak pogled v svoji datoteki.	Hitro oblikovanje aplikacije s pomočjo datoteke storyboard, ki nam omogoča, da so vsi pogledi v eni datoteki.
Dostop do funkcionalnosti mobilne naprave	Dostop preko vtičnikov.	Vgrajeno	Vgrajeno
Cena objave aplikacije	Odvisno od platforme za katero smo prevedli aplikacijo.	Brezplačno	Licenca razvijalca stane 99 USD.
Razvojno okolje	Vsak napredni tekstovni urejevalnik besedila.	Eclipse, Android Studio	Xcode
Govorni vmesnik	Potreben je vtičnik, ki to omogoča.	Ima vgrajenega, vendar podpira malo jezikov, omogoča pa veliko manj funkcionalnosti v primerjavi z govornim vmesnikom na iOS.	Kvaliteten govorni vmesnik z ogromno podporo za jezike.

Tabela 1: Primerjava tehnologij Android, PhoneGap ter iOS pri gradnji naše aplikacije.

Iz tabele lahko sklepamo, da je iOS najprimernejša platforma za doseg naših ciljev. Ima podporo za delo z dokumenti pdf, ima kvaliteten govorni vmesnik, obenem pa je tudi edina platforma, kjer nam je uspelo vstaviti pisavo OpenDyslexic v aplikacijo in v brskalnik. Ovira pri razvoju za iOS je predvsem v licenci razvijalca, ki je plačljiva. Dodaten problem predstavlja distribucija aplikacije, saj je potrebno za to kreirati poseben certifikat. Android je

kvalitetna brezplačna platforma, kjer ni težav pri ceni ali pri distribuciji aplikacije. Toda v primeru zahtev naše aplikacije se je Android izkazal kot neprimerna platforma. Za Android ne obstajajo brezplačne odprtokodne rešitve za razčlenjevanje dokumenta pdf. Android sicer ima govorni vmesnik, vendar ima slabo podporo za jezike, prav tako nam ni uspelo modificirati spletne strani, da bi bila lažje berljiva. Orodje PhoneGap nam sicer nudi hiter razvoj za več platform, vendar pa so aplikacije, razvite s tem orodjem, slabše kvalitete (predvsem v smislu odzivnosti), obenem pa orodje PhoneGap ne nudi vseh funkcionalnosti, ki jih nudijo domorodne aplikacije.

6 Sklepne ugotovitve

Mobilna aplikacija za ljudi z bralnimi težavami je skupen projekt Fakultete za računalništvo in informatiko ter Naravoslovnotehniške fakultete. Na tem projektu sem uspešno sodeloval s študentko Naravoslovnotehniške fakultete, aplikacijo pa nameravamo še nadgraditi. Na Naravoslovnotehniški fakulteti razvijajo novo pisavo, ki bo v pomoč osebam z disleksijo. Pisava bo imela poudarek na slovenskem jeziku, kar bo primernejše za uporabnike aplikacije iz Slovenije. Trenutno je pisava še v nastajanju, tako da v tej verziji aplikacije pisava še ni vključena, bo pa vključena v naslednji verziji aplikacije. Tudi samo aplikacijo smo prilagodili tako, da bo vključitev nove pisave enostavna. Upoštevali bomo tudi povratne informacije uporabnikov, ko bo aplikacija na spletni trgovini App Store. Konkretno predloge bomo upoštevali pri naslednji verziji aplikacije. Trenutno aplikacije ne moremo dostaviti testnim uporabnikom, saj je za to potreben certifikat (t.j. provisioning profile) ter posebna koda posamezne iOS naprave (UDID).

Kot študent Fakultete za računalništvo in informatiko sem aplikacijo izdelal ob pomoči študentke Urške Stariha, ki prihaja z Naravoslovnotehniške fakultete in je v okviru svoje diplomske naloge sodelovala pri zasnovi mobilne aplikacije in oblikovanju uporabniškega vmesnika. Skupaj sva odločala o funkcionalnostih aplikacije ter o njenem izgledu. Moje delo kot bodoči programer je bilo osredotočeno na razvoj aplikacije. Poskusili smo sestaviti aplikacijo za operacijski sistem Android, vendar smo imeli težave tako pri razčlenjevanju dokumenta pdf kot pri prikazu modificirane spletne strani. Žal za Android ne obstaja brezplačen razčlenjevalnik pdf, kar nam je onemogočilo razvoj, obenem pa pojasnjuje, zakaj ima Android tako malo aplikacij, ki bi olajšale branje osebam z disleksijo. Poizkusili smo tudi z orodjem PhoneGap, ki je prav tako imelo svoje omejitve, obenem pa smo imeli težave tudi zaradi pomanjkanja izkušenj s tem orodjem. Če bi nam uspelo sestaviti aplikacijo s tem orodjem, bi aplikacija delovala na operacijskem sistemu Android ter iOS. Potrebno je povedati, da kljub temu, da se ni dalo izvesti zadanih ciljev, smo vseeno pridobili veliko izkušenj predvsem pri uporabi orodja PhoneGap, kjer nismo imeli izkušenj.

Veseli me, da nam je pri operacijskem sistemu iOS uspelo implementirati zadane cilje, in sicer da aplikacija poleg branja e-knjig podpira še ogled spletnih strani z dodatnimi nastavitvam, kar olajša branje osebam z disleksijo. Naša aplikacija bo tudi brezplačno dosegljiva na spletni trgovini App Store, dodali pa bomo vsaj eno posodobitev, ko bo dokončana nova pisava z Naravoslovnotehniške fakultete.

Ob izdelavi te aplikacije sem pridobil veliko novih izkušenj, kar je v današnjem času pomemben dejavnik pri iskanju zaposlitve.

Literatura

- [1] (2014) Knjižnica za delo z dokumenti pdf, z imenom PDF.js. Dosegljivo: <http://mozilla.github.io/pdf.js>.
- [2] (2014) Odprtokodna pisava OpenDyslexic. Dosegljivo: <http://opendyslexic.org>.
- [3] (2014) Operacijski sistem iOS. Dosegljiv: <https://developer.apple.com/library/ios/documentation/Miscellaneous/Conceptual/iPhoneOSTechOverview/Introduction/Introduction.html>.
- [4] (2014) Opis programskega jezika Objective-C. Dosegljivo: https://developer.apple.com/library/mac/documentation/cocoa/conceptual/ProgrammingWithObjectiveC/Introduction/Introduction.html#//apple_ref/doc/uid/TP40011210-CH1-SW1.
- [5] (2014) Opis, kaj je disleksija. Dosegljivo: <http://www.metlika.si/lkm/dislektik1.htm>.
- [6] (2014) Pisava Comic Sans. Dosegljivo: http://en.wikipedia.org/wiki/Comic_Sans.
- [7] (2014) Predstavitev PhoneGap tehnologije. Dosegljivo: <http://tehnika.telekom.si/novice/phonegap-s-spletnimi-tehnologijami-do-mobilnih-aplikacij>.
- [8] (2014) Predstavitev pisav za disleksike. Dosegljivo: <http://www.dyslexia-reading-well.com/dyslexia-font.html>.
- [9] (2014) Predstavitev razvojnega orodja Xcode. Dosegljivo: <https://developer.apple.com/xcode>.
- [10] (2014) Prikaz spletne strani z dodajanjem prilagojene slogovne predloge. Dosegljivo na: <http://stackoverflow.com/questions/4950729/rendering-html-in-a-webview-with-custom-css>.
- [11] (2014) Prikaz uporabe govornega vmesnika (angl. Text to Speech) za iOS. Dosegljivo: <http://useyourloaf.com/blog/2014/01/08/synthesized-speech-from-text.html>.
- [12] (2014) Priporočila za osebe z disleksijo. Dosegljivo: <http://www.bdadyslexia.org.uk/about-dyslexia/further-information/dyslexia-style-guide.html>.
- [13] (2014) Razčlenjevalnik za dokumente pdf. Dosegljivo: <http://fastpdfkit.com>.
- [14] (2014) Uvoz dokumenta v aplikacijo. Dosegljivo: <http://mobiforge.com/design-development/importing-exporting-documents-ios>.

- [15] Vaje za učenje razvoja z Xcodom ter iOSom: Dosegljivo:
https://developer.apple.com/library/ios/referencelibrary/GettingStarted/RoadMapiOS/FirstTutorial.html#//apple_ref/doc/uid/TP40011343-CH3-SW1.
- [16] (2014) Vtičniki, ki jih uporablja PhoneGap. Dosegljivo:
http://docs.phonegap.com/en/3.5.0/cordova_plugins_pluginapis.md.html#Plugin%20APIs.
- [17] (2014) Zgodovina programskega jezika Objective-C. Dosegljivo:
<http://en.wikipedia.org/wiki/Objective-C>.